

In this analysis is performed to get the insights from the data. Below mentioned are the queries through which analysis is performed.

Query Query History

```
1 Q1: Senior most employee based on job title
2
3 select first_name , last_name from employee order by levels desc limit 1
```

Data Output Messages Notifications



	first_name character (50)		last_name character (50)	
1	Mohan	...	Madan	...

Query Query History

```
1 Q2: Which country have the most Invoices?
2
3 select billing_country, count(*) as c
4 from invoice
5 group by billing_country
6 order by c desc
```

Data Output Messages Notifications



	billing_country character varying (30)	c bigint
1	USA	131
2	Canada	76
3	Brazil	61
4	France	50
5	Germany	41
6	Czech Republic	30
7	Portugal	29
8	United Kingdom	28
9	India	21
10	Chile	13

Query Query History

```
1 Q3: What are top 3 values of total invoice
2
3 select total from invoice
4 order by total desc limit 3
```

Data Output Messages Notifications



	total double precision	
1	23.759999999999998	
2	19.8	
3	19.8	

Query Query History

```
1 Q4: Which city has the best customers? We would like to throw a promotional Music Festival
2 in the city we made the most money. Write a query that returns one city that has the highest
3 sum of invoice totals. Return both the city name & sum of all invoice totals
4
5 select billing_city, SUM(total)as invoice_Total
6 from invoice
7 group by billing_city
8 order by invoice_Total desc limit 1
```

Data Output Messages Notifications



	billing_city character varying (30)	invoice_total double precision
1	Prague	273.240000000000007

Query Query History

```
1 Q5: Who is the best customer? The customer who has spent the most money will be declared the best
2 customer. Write a query that returns the person who has spent the most money.
3
4 select customer.customer_id, customer.first_name, customer.last_name, SUM(invoice.total) as total
5 from customer
6 inner join invoice on customer.customer_id= invoice.customer_id
7 group by customer.customer_id
8 order by total desc limit 1
```

Data Output Messages Notifications

	customer_id (FK) integer	first_name character (50)	last_name character (50)	total double precision
1	5	R	Madhav	144.54000000000002

Query Query History

```
1 Q6: Write query to return the email, first name, last name & Genre of all Rock Music listeners.
2 Return your ordered alphabetically by email starting with A
3
4 select distinct email, first_name, last_name
5 from customer
6 join invoice on customer.customer_id= invoice.customer_id
7 join invoice_line on invoice.invoice_id= invoice_line.invoice_id
8 where track_id IN(
9     select track_id from track
10     join genre on track.genre_id = genre.genre_id
11     where genre.name like 'Rock'
12 )
13 order by email;
```

Data Output Messages Notifications

	email character varying (50)	first_name character (50)	last_name character (50)
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	alero@uol.com.br	Alexandre	Rocha
3	astrid.gruber@apple.at	Astrid	Gruber
4	bjorn.hansen@yahoo.no	Bjorn	Hansen
5	camille.bernard@yahoo.fr	Camille	Bernard
6	daan.peeters@apple.be	Daan	Peeters
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez
8	dmiller@comcast.com	Dan	Miller
9	dominiquelafevre@gmail.c...	Dominique	Lefebvre

Query Query History

```
1 Q7. Lets invite the artists who have written the most rock music in our dataset. Write a query
2 that returns the Artist name and total track count of the top 10 rock bands
3
4 Select artist.artist_id,artist.name,COUNT(artist.artist_id)AS number_of_songs
5 from track
6 JOIN album on album.album_id = track.album_id
7 join artist on artist.artist_id = album.artist_id
8 join genre on genre.genre_id = track.genre_id
9 where genre.name LIKE 'Rock'
10 group by artist.artist_id
11 order by number_of_songs DESC
12 limit 10;
```

Data Output Messages Notifications			
	artist_id [PK] character varying (50)	name character varying (120)	number_of_songs bigint
1	22	Led Zeppelin	114
2	150	U2	112
3	58	Deep Purple	92
4	90	Iron Maiden	81
5	118	Pearl Jam	54
6	152	Van Halen	52
7	51	Queen	45
8	142	The Rolling Stones	41
9	76	Creedence Clearwater Revival	40

Query Query History

```
1 Q8.-- Return all the track names that have a song length longer than the average song length.
2 -- Return the name and Milliseconds for each track. Order by the song length with the
3 -- longest songs listed first.
4
5 select name, milliseconds
6 from track
7 where milliseconds > (
8 Select avg(milliseconds) AS avg_track_length
9 from track)
10 order by milliseconds desc;
11
12
13
```

Data Output Messages Notifications		
	name character varying (150)	milliseconds integer
1	Occupation / Precipice	5286953
2	Through a Looking Glass	5088838
3	Greetings from Earth, Pt. 1	2960293
4	The Man With Nine Lives	2956998
5	Battlestar Galactica, Pt. 2	2956081
6	Battlestar Galactica, Pt. 1	2952702
7	Murder On the Rising Star	2935894
8	Battlestar Galactica, Pt. 3	2927802
9	Take the Celestra	2927677

```
Query Query History
1 Q9. --Find how much amount spent by each customer on artists? Write a query to return customer
2   --name, artist name and total spent
3
4 With best_selling_artist AS(
5     select artist.artist_id as artist_id, artist.name as artist_name,
6     SUM(invoice_line.unit_price*invoice_line.quantity)AS total_sales
7     FROM invoice_line
8     JOIN track ON track.track_id = invoice_line.track_id
9     JOIN album ON album.album_id = track.album_id
10    JOIN artist ON artist.artist_id = album.artist_id
11    GROUP BY 1
12    ORDER BY 3 DESC
13    LIMIT 1
14 )
15 SELECT c.customer_id, c.first_name, c.last_name, bsa.artist_name,
16 SUM(il.unit_price*il.quantity)AS amount_spent
17 FROM invoice i
18 JOIN customer c ON c.customer_id = i.customer_id
19 JOIN invoice_line il ON il.invoice_id = i.invoice_id
20 JOIN track t ON t.track_id = il.track_id
21 JOIN album alb ON alb.album_id = t.album_id
22 JOIN best_selling_artist bsa ON bsa.artist_id = alb.artist_id
23 GROUP BY 1,2,3,4
24 ORDER BY 5 DESC;
25
```

1 Data Output Messages Notifications

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

	customer_id integer	first_name character (50)	last_name character (50)	artist_name character varying (120)	amount_spent double precision
1	46	Hugh	O'Reilly	Queen	27.719999999999985
2	38	Niklas	Schröder	Queen	18.81
3	3	François	Tremblay	Queen	17.82
4	34	João	Fernandes	Queen	16.830000000000002
5	53	Phil	Hughes	Queen	11.88
6	41	Marc	Dubois	Queen	11.88
7	47	Lucas	Mancini	Queen	10.89
8	33	Ellie	Sullivan	Queen	10.89
9	20	Dan	Miller	Queen	3.96
10	5	R	Madhav	Queen	3.96
11	23	John	Gordon	Queen	2.9699999999999998
12	54	Steve	Murray	Queen	2.9699999999999998
13	31	Martha	Silk	Queen	2.9699999999999998
14	16	Frank	Harris	Queen	1.98
15	17	Jack	Smith	Queen	1.98
16	24	Frank	Ralston	Queen	1.98
17	30	Edward	Francis	Queen	1.98
18	35	Madalena	Sampaio	Queen	1.98
19	36	Hannah	Schneider	Queen	1.98
20	11	Alexandre	Rocha	Queen	1.98
21	8	Daan	Peeters	Queen	1.98
22	42	Wyatt	Girard	Queen	1.98
23	44	Terhi	Hämäläinen	Queen	1.98
24	1	Luís	Gonzalez	Queen	1.98

Q10. -- We want to find out the most popular music Genre for each country (We determine the most popular genre -- as the genre with the highest amount of purchases). Write a query that returns each country along with -- the top Genre. For countries where the maximum number of purchases is shared return all Genres

With RECURSIVE

```
sales_per_country AS(
  SELECT COUNT(*) AS purchases_per_genre, customer.country, genre.name, genre.genre_id
  FROM invoice_line
  JOIN invoice ON invoice.invoice_id = invoice_line.invoice_id
  JOIN customer ON customer.customer_id = invoice.customer_id
  JOIN track ON track.track_id = invoice_line.track_id
  JOIN genre ON genre.genre_id = track.genre_id
  GROUP BY 2,3,4
  ORDER BY 2
),
max_genre_per_country AS (SELECT MAX(purchases_per_genre) AS max_genre_number, country
  FROM sales_per_country
  GROUP BY 2
  ORDER BY 2)
SELECT sales_per_country.*
FROM sales_per_country
JOIN max_genre_per_country ON sales_per_country.country = max_genre_per_country.country
WHERE sales_per_country.purchases_per_genre = max_genre_per_country.max_genre_number
```

Data Output Messages Notifications

	purchases_per_genre bigint	country character varying (50)	name character varying (120)	genre_id character varying (50)
1	17	Argentina	Alternative & Punk	4
2	34	Australia	Rock	1
3	40	Austria	Rock	1
4	26	Belgium	Rock	1
5	205	Brazil	Rock	1
6	333	Canada	Rock	1
7	61	Chile	Rock	1
8	143	Czech Republic	Rock	1
9	24	Denmark	Rock	1
10	46	Finland	Rock	1
11	211	France	Rock	1
12	194	Germany	Rock	1
13	44	Hungary	Rock	1
14	102	India	Rock	1
15	72	Ireland	Rock	1
16	35	Italy	Rock	1
17	33	Netherlands	Rock	1
18	40	Norway	Rock	1
19	40	Poland	Rock	1
20	108	Portugal	Rock	1
21	46	Spain	Rock	1
22	60	Sweden	Rock	1
23	166	United Kingdom	Rock	1
24	561	USA	Rock	1

Q11. --Write a query that determines the customer that has spent the most on music for each country. Write  
 --a query that returns the country along with the top customer and how much they spent. For countries  
 --where the top amount spent is shared, provide all customers who spent this amount.

With RECURSIVE

```
customer_with_country AS (
  SELECT customer.customer_id,first_name,last_name,billing_country,SUM(total) AS total_spending
  FROM invoice
  JOIN customer ON customer.customer_id = invoice.customer_id
  GROUP BY 1,2,3,4
  ORDER BY 1,5 DESC),

country_max_spending AS (
  SELECT billing_country,MAX(total_spending) AS max_spending
  FROM customer_with_country
  GROUP BY billing_country)

SELECT cc.billing_country,cc.total_spending,cc.first_name,cc.last_name,cc.customer_id
FROM customer_with_country cc
JOIN country_max_spending ms
ON cc.billing_country = ms.billing_country
WHERE cc.total_spending = ms.max_spending
ORDER BY 1;|
```

Data Output Messages Notifications

	billing_country character varying (30)	total_spending double precision	first_name character (50)	last_name character (50)	customer_id integer
1	Argentina	39.6	Diego	Gutiérrez	56
2	Australia	81.18	Mark	Taylor	55
3	Austria	69.3	Astrid	Gruber	7
4	Belgium	60.389999999999999	Daan	Peeters	8
5	Brazil	108.899999999999998	Luís	Gonçalves	1
6	Canada	99.99	François	Tremblay	3
7	Chile	97.020000000000001	Luis	Rojas	57
8	Czech Republic	144.540000000000002	R	Madhav	5
9	Denmark	37.619999999999999	Kara	Nielsen	9
10	Finland	79.2	Terhi	Hämäläinen	44
11	France	99.99	Wyatt	Girard	42
12	Germany	94.050000000000001	Fynn	Zimmermann	37
13	Hungary	78.21	Ladislav	Kovács	45
14	India	111.869999999999999	Manoj	Pareek	58
15	Ireland	114.839999999999997	Hugh	O'Reilly	46
16	Italy	50.49	Lucas	Mancini	47
17	Netherlands	65.34	Johannes	Van der Berg	48
18	Norway	72.270000000000001	Bjørn	Hansen	4
19	Poland	76.229999999999999	Stanislaw	Wójcik	49
20	Portugal	102.960000000000001	João	Fernandes	34
21	Spain	98.01	Enrique	Muñoz	50
22	Sweden	75.24	Joakim	Johansson	51
23	United Kingdom	98.01	Phil	Hughes	53
24	USA	98.01	Jack	Smith	17